Amendments to the Claims

1. (Currently Amended) A method of synchronizing a plurality of processors of a multi-processor computer system on a synchronization point, said method comprising:

creating a circular reference arrangement for said plurality of processors, wherein a first processor from among said plurality of processors is designated said lead processor and a second processor from among said plurality of processors is designated said tail processor, and wherein said lead processor is adjacent to said tail processor in said circular reference arrangement;

triggering a first set of processors to enter an entry holding loop in response to said first set of processors encountering said synchronization point before [[a]] said lead processor associated with said plurality of processors;

triggering said first set of processors to enter an exit holding loop in response to said lead processor encountering said synchronization point, said first set of processors entering said exit holding loop in a cascading manner starting with said lead processor following a sequence established by said circular reference arrangement, with each processor of said first set of processors being triggered by its immediate predecessor in said sequence; and

triggering said plurality of processors to leave said exit holding loop in response to [[a]] said tail processor associated with said plurality of processors encountering said synchronization point, said plurality of processors leaving said exit holding loop in a cascading manner starting with said tail processor following said sequence established by said circular reference arrangement, with each processor of said plurality of processors being triggered by its immediate predecessor in said sequence.

2. (Cancelled).

3. (Currently Amended) The method of claim [[2]] 1 wherein said creating said circular reference arrangement for said plurality of processors, wherein said first processor is designated said lead processor and said second processor is designated said tail processor, and wherein said lead processor is adjacent to said tail processor in said circular reference arrangement, further comprises:

creating said circular reference arrangement for said plurality of processors, wherein said first processor is designated said lead processor and said second processor is designated said tail processor, and wherein said lead processor is adjacent to said tail processor in said circular reference arrangement, said circular reference arrangement representing a circular linked list.

4. (Cancelled).

5. (Currently Amended) The method of claim [[4]] 1 wherein said triggering said plurality of processors to leave said exit holding loop in response to said tail processor encountering said synchronization point further comprises:

triggering said plurality of processors to leave said exit holding loop in response to said tail processor encountering said synchronization point, said plurality of processors leaving said exit holding loop in a cascading manner starting with said tail processor following said sequence established by said circular reference arrangement, with each processor of said plurality of processors being triggered by its immediate predecessor in said sequence.

6. (Cancelled).

7. (Previously Presented) The method of claim 1, further comprising:

employing a first external interrupt mechanism associated with each of said first set of processors.

8. (Previously Presented) The method of claim 7 further comprising:

writing to hard physical addresses of each of said first set of processors.

9. (Previously Presented) The method of claim 7, further comprising:

employing a second external interrupt mechanism associated with each of said plurality of processors.

10. (Previously Presented) The method of claim 9 further comprising:

writing to hard physical addresses of each of said plurality of processors.

11. (Previously Presented) The method of claim 1 wherein said triggering said first set of processors to enter said exit holding loop in response to said lead processor encountering said synchronization point further comprises:

triggering said first set of processors to enter said exit holding loop in response to said lead processor encountering said synchronization point utilizing a masked interrupt approach.

12. (Currently Amended) An article of manufacture comprising a program storage medium having computer readable code embodied therein, wherein executing said computer readable code causes a computer to implement a method of synchronizing a plurality of processors on a synchronization point, said method comprising:

creating a circular reference arrangement for said plurality of processors, wherein a first processor from among said plurality of processors is designated said lead processor and a second processor from among said plurality of processors is designated said tail processor, and wherein said lead processor is adjacent to said tail processor in said circular reference arrangement;

triggering a first set of processors to enter an entry holding loop in response to said first set of processors encountering said synchronization point before [[a]] said lead processor associated with said plurality of processors;

triggering said first set of processors to enter an exit holding loop in response to said lead processor encountering said synchronization point, said first set of processors entering said exit holding loop in a cascading manner starting with said lead processor following a sequence established by said circular reference arrangement, with each processor of said first set of processors being triggered by its immediate predecessor in said sequence; and

triggering said plurality of processors to leave said exit holding loop in response to [[a]] said tail processor encountering said synchronization point, said plurality of processors leaving said exit holding loop in a cascading manner starting with said tail processor following said sequence established by said circular reference arrangement, with each processor of said plurality of processors being triggered by its immediate predecessor in said sequence.

13. (Cancelled).

14. (Currently Amended) The article of manufacture of claim [[13]] 12 wherein said creating said circular reference arrangement for said plurality of processors, wherein said first processor is designated said lead processor and said second processor is designated said tail processor, and wherein said lead processor is adjacent to said tail processor in said circular reference arrangement, further comprises:

creating said circular reference arrangement for said plurality of processors, wherein said first processor is designated said lead processor and said second processor is designated said tail processor, and wherein said lead processor is adjacent to said tail processor in said circular reference arrangement, said circular reference arrangement representing a circular linked list.

15. (Cancelled).

16. (Cancelled).

17. (Cancelled).

18. (Previously Presented) The article of manufacture of claim 12 wherein said method further comprises:

employing a first external interrupt mechanism associated with each of said first set of processors.

19. (Previously Presented) The article of manufacture of claim 18 wherein said method further comprises:

writing to hard physical addresses of each of said first set of processors.

20. (Previously Presented) The article of manufacture of claim 18 wherein said method further comprises:

employing a second external interrupt mechanism associated with each of said plurality of processors.

21. (Previously Presented) The article of manufacture of claim 20 wherein said method further comprises:

writing to hard physical addresses of each of said plurality of processors.

22. (Previously Presented) The article of manufacture of claim 12 wherein said method further comprises:

receiving trigger signals using a masked interrupt approach.

23. (Currently Amended) A method for synchronizing a plurality of processors of a multi-processor computer system on a synchronization point in instantiations of a computer program, comprising:

implementing a circular reference arrangement representing a circular linked list for said plurality of processors, each of said plurality of processors having an immediately preceding processor and an immediately succeeding processor, one of said plurality of processors being designated said lead processor, another one of said processors being designated said tail processor, said lead processor immediately succeeding said tail processor in said circular reference arrangement;

keeping a first set of processors in an entry holding loop when said processors of said first set of processors reach said synchronization point, said first set of processors representing said plurality of processors except said lead processor;

cascade triggering along said circular reference arrangement said first set of processors, using a lead processor of said plurality of processors when said lead processor encounters said synchronization point, to enter an exit holding loop, said cascade triggering said first set of processors being performed without accessing a shared memory area of said multi-processor system;

keeping said lead processor in said exit holding loop; and thereafter

cascade triggering along said circular reference arrangement said plurality of processors, using a tail processor of said plurality of processors when said tail processor encounters said synchronization point, to leave said exit holding loop, said cascade triggering said plurality of processors being performed without accessing said shared memory area of said multi-processor system.

24. (Cancelled).

25. (Original) The method of claim 23 wherein said cascade triggering said first set of processors employs a first external interrupt mechanism associated with each of said first set of processors.

26. (Original) The method of claim 25 wherein said cascade triggering said first set of processors includes writing to hard physical addresses of each of said first set of processors.

27. (Original) The method of claim 25 wherein said cascade triggering said plurality of processors employs a second external interrupt mechanism associated with each of said plurality of processors.

28. (Original) The method of claim 27 wherein said cascade triggering said plurality of processors includes writing to hard physical addresses of each of said plurality of processors.

29. (Original) The method of claim 23 wherein said cascade triggering employs a masked interrupt approach.